

**REMARKS**

The application has been reviewed and amended in light of the Office Action dated January 21, 2004. Claims 1-5 and 11-12 are currently pending in this application. Claims 6-10, 13, and 14 have been cancelled hereby, without prejudice or disclaimer, as required by the Office Action. Of the claims presently under consideration, claims 1 and 11 are in independent form. It is submitted that no new matter has been added and no new issues have been raised by the present amendment.

Claims 1-5, 11, and 12 have been rejected under 35 U.S.C. § 102(e) as allegedly being anticipated by U.S. Patent No. 6,085,188 to Bachmann et al.

Applicant has carefully considered the Examiner's comments and the cited art, and respectfully submits that independent claim 1 is patentably distinct from the cited art for at least the following reasons.

Independent claim 1 relates to a method for improving the operational performance of a database system, the method comprising: determining whether an instruction or operation adds information or removes information from the database system, wherein for an add operation, the information is first added to an 'out' table, and wherein for a remove operation, the information is first removed from an 'in' table.

Bachmann et al., as understood by Applicant, relates to a method of hierarchical LDAP searching in an LDAP directory service having a relational database management system (DBMS) as a backing store. Entries in a naming hierarchy are mapped into first and second relational tables: a parent table and a descendant table. These tables are used to filter lists of entries returned from a search to ensure that only entries within a given search scope are retained for evaluation.

The Office Action cites col. 6, ln. 47 to col. 7, ln. 15 and Figs. 7-8 of Bachmann et al. as allegedly disclosing determining whether an instruction or operation adds information or removes information from the database system, wherein for an add operation, the information is first added to an 'out' table, and wherein for a remove operation, the information is first removed from an 'in' table (see Office Action, p. 3, lns. 5-11). The Office Action further states that the descendant table of Bachmann et al. corresponds to an 'in' table, and that the parent table of Bachmann et al. corresponds to an 'out' table (see id.). Applicant respectfully disagrees.

As understood by Applicant, Bachmann et al. discloses mapping entries in a naming hierarchy having a plurality of entries each represented by a unique identifier (EID) into parent and descendent relational tables (see Bachmann et al., col. 2, lns. 30-35). The tables are used to filter lists of entries returned from a search to ensure that only entries within a given search scope are retained for evaluation (see id., lns. 59-65).

For example, the parent table is used during a LDAP one level search, and the descendent table is used during a LDAP subtree search. Use of the parent or descendent table obviates recursive queries through the naming directory (see id.).

Fig. 5 of Bachmann et al. illustrates the LDAP naming hierarchy including a number of entries or nodes, with each entry or node represented by an EID (see id., col. 5, lns. 12-14). Figs. 6A-6B of Bachmann et al. illustrate the mapping of the LDAP directory service naming hierarchy into preferably a pair of relational tables (see id., col. 4, ln. 65 to col. 5, ln. 9).

As understood by Applicant, the parent table of Bachmann et al. summarizes parent-child relationships in the directory service naming hierarchy (see id., col. 2, lns. 45-46).

In the parent table, the EID field is the unique identifier of an entry in the LDAP naming hierarchy, and the PEID field is the unique identifier of the parent entry in the naming hierarchy (see id., col. 5, lns. 51-59).

The descendant table of Bachmann et al., as understood by Applicant, summarizes ancestor-descendant relationships in the directory service naming hierarchy (see id., col. 2, lns. 57-58). In the descendent table, the AEID field is the unique identifier of an ancestor LDAP entry in the LDAP naming hierarchy, and the DEID field is the unique identifier of the descendent LDAP entry (see id., col. 5, lns. 51-59).

In contrast, as stated in the specification of the present application, “[t]he idea behind using an ‘in’ table and ‘out’ table structure, is that a search can be conducted on an ‘in’ table, a search table for example, and the results of that search can be based on an ‘out’ table, an entry table ... ” (see specification of the present application, p. 3, lns. 27-29).

Additionally, the specification states “... when an add entry operation is performed, information is added to the entry table first, that is to the ‘out’ table, such that the information is not visible initially. The information is then added to the search table, that is the ‘in’ table, such that the information is visible and searchable so that all corresponding information can be retrieved. Thus, according to this embodiment of the present application, information to be added to a database is first prebuilt in a non-visible table before the information is made visible (see id., p. 5, lns. 17-28). “In other words, as rows are added to the ‘in’ table, the entry gradually becomes visible, and any search (on an ‘in’ table attribute) if found for a partially visible entry, the complete entry will be read” (see id.). Of course, the claims are not limited to the disclosed embodiments.

It is respectfully submitted that there is no indication, either in the cited reference or in the Office Action, that the parent and descendent tables of Bachmann et al. disclose or relate to the 'in' and 'out' tables of the present disclosure and as recited in independent claim 1.

It is further submitted that, as understood by Applicant, the parent and descendent tables of Bachmann et al. do not function as or correspond to 'in' and 'out' tables, respectively.

It is respectfully submitted that Bachmann et al. does not disclose or suggest a method for improving the operational performance of a database system, the method comprising: determining whether an instruction or operation adds information or removes information from the database system, wherein for an add operation, the information is first added to an 'out' table, and wherein for a remove operation, the information is first removed from an 'in' table, as recited in independent claim 1.

Accordingly, for at least the reasons set forth above, independent claim 1, and the claims depending therefrom, are believed to be patentable over the cited art. Independent claim 11 is believed to be patentable over the cited art for at least similar reasons. Independent claim 11, and the claims depending therefrom, are believed to be patentable over the cited art for at least similar reasons.

Withdrawal of the rejection of claims 1-5, 11, and 12 under 35 U.S.C. § 102(e) is respectfully requested.

Entry of this response is earnestly solicited, and it is respectfully submitted that this response raises no new issues requiring further consideration and/or search, because the functional aspects of the invention have merely been clarified in the above remarks.

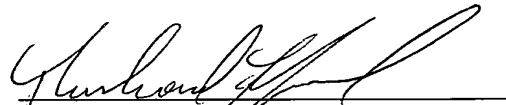
The Office is hereby authorized to charge any additional fees that may be required in connection with this response and to credit any overpayment to our Deposit Account No. 03-3125.

If a petition for an extension of time is required to make this response timely, this paper should be considered to be such a petition, and the Commissioner is authorized to charge the requisite fees to our Deposit Account No. 03-3125.

If a telephone interview could advance the prosecution of this application, the Examiner is respectfully requested to call the undersigned attorney.

Entry of this amendment and allowance of this application are respectfully requested.

Respectfully submitted,

A handwritten signature in black ink, appearing to read 'Richard F. Jaworski', is written over a horizontal line.

RICHARD F. JAWORSKI  
Reg. No.33,515  
Attorney for Applicants  
Cooper & Dunham LLP  
Tel.: (212) 278-0400